UNITED STATES PATENT AND TRADEMARK OFFICE

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.'	CONFIRMATION NO.
10/079,255	02/20/2002	Thomas Kunzler	KUNZ/0001	6099

24945	7590	01/13/2005	

EXAMINER
FOWLKES, ANDRE R

STREETS & STEELE
13831 NORTHWEST FREEWAY
SUITE 355
HOUSTON, TX 77040

ART UNIT	PAPER NUMBER
2122	

DATE MAILED: 01/13/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

PTO-90C (Rev. 10/03)

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE <u>3</u> MONTH(S) FROM
THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed
  after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133).
  Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any
  earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on <u>20 May 2002</u>.

2a)☐ This action is **FINAL**.   2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is
closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) <u>1-20</u> is/are pending in the application.

    4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) <u>1-20</u> is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on <u>2/20/02</u> is/are: a)☒ accepted or b)☐ objected to by the Examiner.

    Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

    Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

    a)☐ All   b)☐ Some * c)☐ None of:

      1.☐ Certified copies of the priority documents have been received.

      2.☐ Certified copies of the priority documents have been received in Application No. _____.

      3.☐ Copies of the certified copies of the priority documents have been received in this National Stage
application from the International Bureau (PCT Rule 17.2(a)).

    * See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1)☒ Notice of References Cited (PTO-892)

2)☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3)☒ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
    Paper No(s)/Mail Date <u>4/29/02</u>.

4)☐ Interview Summary (PTO-413)
    Paper No(s)/Mail Date. _____ .

5)☐ Notice of Informal Patent Application (PTO-152)

6)☐ Other: _____ .

DETAILED ACTION

1.   Claims 1-20 are pending.

Claim Rejections - 35 USC § 102

2.   The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that

form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

3.   Claims 1-20 are rejected under 35 U.S.C. 102(e) as being anticipated by Gunter

et al., (Gunter), U.S. Patent Application Publication No. 2001/0011370.

        As per claim 1, Gunter discloses **a method for automatically generating**

flowcharting instructions from a source code (¶ 34:1-2, "server unit 20 translates

code 64 into flow chart 68"), **comprising:**

        - loading source code statements into statement records of a data

structure, wherein each statement record contains only one of the source code

statements (¶ 35:1-5, "In accordance with one embodiment of the invention server unit

20 generates a DOT file (i.e. data structure) that contains the (source code statement

records expressed as)... nodes and edges, in DOT syntax, relating to the graph

representations for flow chart 68. This DOT file is then provided to graphic drawing

unit"),

  - **identifying each branch statement contained in the statement records** (¶

34:7-11, "The condition "while" in code 64 is translated to a diamond box 76. As

illustrated two branches or edges 78 and 80 are also generated relating to the "YES"

and "NO" conditions associated with "while" construct of code 64.", and fig. 2B),

  - **determining one or more branch destinations for each of the branch**

statements (¶ 34:7-11, "The condition "while" in code 64 is translated to a diamond box

76. As illustrated two branches or edges 78 and 80 are also generated relating to the

"YES" and "NO" conditions associated with "while" construct of code 64.", and fig. 2B),

  - **storing the one or more branch destinations in one or more destination**

records of the data structure, wherein the one or more destination records

correspond to the statement record of the branch statement (¶ 34:7-11, "The

condition "while" in code 64 is translated to a diamond box 76. As illustrated two

branches or edges 78 and 80 are also generated relating to the "YES" and "NO"

conditions associated with "while" construct of code 64.", and fig. 2B),

  - **identifying a statement type for each statement contained in the statement**

records (¶ 21:3-7, "Program constructs include individual operations that together

accomplish a software task, such as "begin" statements, "end" statements, "if"

statements, "if-then-else" statements, "wait" statements and "assignment" statements.").

As per claim 2, the rejection of claim 1 is incorporated and further, Gunter discloses **maintaining a record management system of statement conventions comprising defined branch statement records and alternative branch statement records, wherein each alternative branch statement record is associated with one defined branch statement record** (¶ 34:7-11, "The condition "while" in code 64 is translated to a diamond box 76. As illustrated two branches or edges 78 and 80 are also generated relating to the "YES" (i.e. branch statement record) and "NO" (i.e. alternative branch statement) conditions associated with "while" construct of code 64.", and fig. 2B)

As per claim 3, the rejection of claim 2 is incorporated and further, Gunter discloses that **the record management system of statement conventions is selected from a database, a spreadsheet, an array, a set of coded rules in an applications program and combinations thereof** (¶ 66:5-8, "Thus, a data structure in file system unit 16 is created that contains information relating to all the edges (and nodes) in the flow chart, including the label on each edge").

As per claim 4, the rejection of claim 2 is incorporated and further, Gunter discloses **finding any alternative branch statements in the statement records; and replacing each of the found alternative branch statements with the associated defined statement** (¶ 34:7-11, "The condition "while" in code 64 is translated (i.e. replaced with) a diamond box 76. As illustrated two branches or edges 78 and 80 are

also generated relating to the "YES" (i.e. branch statement record) and "NO" (i.e.

alternative branch statement) conditions associated with "while" construct of code 64").

As per claim 5, the rejection of claim 1 is incorporated and further, Gunter

discloses that **the source code statements are written in a computer programming**

language selected from FORTRAN, COBOL, C/C++, PASCAL, Visual Basic, and

PERL (see fig. 2A, items 60 and 62 and associated text (e.g. ¶ 32:1-38:10)).

As per claim 6, the rejection of claim 1 is incorporated and further, Gunter

discloses that **the statement types are selected from decision, input, output,**

comment, computational, arithmetic, data, type, logic, start, loop, transfer and

end (¶ 21:3-7, "Program constructs include individual operations that together

accomplish a software task, such as "begin" statements, "end" statements, "if"

statements, "if-then-else" statements, "wait" statements and "assignment" statements",

and fig. 2A, items 60 and 62 and associated text (e.g. ¶ 32:1-38:10)).

As per claim 7, the rejection of claim 1 is incorporated and further, Gunter

discloses that **the statement types are selected from executable statements, non-**

executable statements, and comment statements (¶ 21:3-7, "Program constructs

include individual operations that together accomplish a software task, such as "begin"

statements, "end" statements, "if" statements, "if-then-else" statements, "wait"

statements and "assignment" statements", and fig. 2A, items 60 and 62 and associated

text (e.g. ¶ 32:1-38:10)).

As per claim 8, the rejection of claim 1 is incorporated and further, Gunter

discloses that **the statement types are selected from chain statements and branch**

statements (¶ 21:3-7, "Program constructs include individual operations that together

accomplish a software task, such as "begin" statements, "end" statements, "if"

statements, "if-then-else" statements, "wait" statements and "assignment" statements",

and fig. 2A, items 60 and 62 and associated text (e.g. ¶ 32:1-38:10)).

As per claim 9, the rejection of claim 1 is incorporated and further, Gunter

discloses **maintaining a record management system of statement classifications**

comprising, for each source code statement type, a statement key, a statement

format, destination locations, branch connector labels, a flowchart shape, and

combinations thereof (¶ 66:5-8, "Thus, a data structure in file system unit 16 is

created that contains information relating to all the edges (and nodes) in the flow chart,

including the label on each edge").

As per claim 10, the rejection of claim 9 is incorporated and further, Gunter

discloses that **the statement key comprises a word, a phrase, punctuation marks,**

other symbols, and combinations thereof (¶ 21:3-7, "Program constructs include

individual operations that together accomplish a software task, such as "begin"

statements, "end" statements, "if" statements, "if-then-else" statements, "wait"

statements and "assignment" statements (i.e. statement keys)").

As per claim 11, the rejection of claim 9 is incorporated and further, Gunter

discloses that **the record management system of statement classifications is**

selected from a database, a spreadsheet, an array, a set of coded rules in an

applications program, and combinations thereof (¶ 66:5-8, "Thus, a data structure in

file system unit 16 is created that contains information relating to all the edges in the

flow chart, including the label on each edge").

As per claim 12, the rejection of claim 1 is incorporated and further, Gunter

discloses that the step of identifying a statement type for each of the source code

statements contained in the statement records further comprises:

- matching a statement key with each of the source code statements, wherein the

statement key is associated with one statement type, and wherein the statement

key comprises a word, a phrase, punctuation marks, other symbols, and

combinations thereof, (¶ 21:3-7, "Program constructs include individual operations

that together accomplish a software task, such as "begin" statements, "end" statements,

"if" statements, "if-then-else" statements, "wait" statements and "assignment"

statements (i.e. statement keys)"), **and recording in the data structure the statement**

type for each statement in the statement records (¶ 66:5-8, "Thus, a data structure

in file system unit 16 is created that contains information relating to all the edges in the

flow chart, including the label on each edge").

As per claim 13, the rejection of claim 12 is incorporated and further, Gunter

discloses **assigning a flowchart shape to each statement of the source code,**

wherein each flowchart shape is associated with the statement type of each

statement of the source code, (¶ 34:1-8, "Thus, as illustrated in FIGS. 2a and 2b,

server unit 20 translates code 64 into flow chart 68. For example, the first construct

"begin" in code 64 is translated into a box 70 containing a text "0:START" in flow chart

68. Similarly, the assignments of variables "y1" and "y2" are translated to assignment

boxes 72 and 74 containing corresponding assignment texts. The condition "while" in

code 64 is translated to a diamond box 76. As illustrated two branches or edges 78 and

80 are also generated relating to the "YES" and "NO" conditions associated with "while"

construct of code 64. Condition 82 of code 64 is translated to diamond box 84 in flow

chart 66"), **and storing the assigned flowchart shape in a shape record in the data**

structure corresponding to the statement record (¶ 66:5-8, "Thus, a data structure

in file system unit 16 is created that contains information relating to all the edges (and

nodes) in the flow chart, including the label on each edge").

As per claim 14, the rejection of claim 1 is incorporated and further, Gunter

discloses that **the data structure is selected from a spreadsheet, a database, a**

linked list, an array and combinations thereof (¶ 66:5-8, "Thus, a data structure in

file system unit 16 is created that contains information relating to all the edges in the

flow chart, including the label on each edge").

As per claim 15, the rejection of claim 1 is incorporated and further, Gunter

discloses that **the data structure is created in a computer language selected from

COBOL, C, PASCAL and C/C.sup.++** (¶ 33:3-5, "In accordance with one embodiment

of the invention, code 64 is written in a language that employs Pascal-like syntax,

although the invention is not limited in scope in that respect and other languages may

be employed in generating code").

As per claim 16, the rejection of claim 1 is incorporated and further, Gunter

discloses **the step of loading source code statements into statement records of a

data structure further comprises:**

- assigning a unique statement number to each statement record (fig. 2B, all

statements are assigned a unique statement number, e.g. item 72, "1: y:1=x", and

associated text, (¶ 32:1-38:10)),

**- storing each unique statement number as a statement number record

associated with each statement record in the data structure** (¶ 66:5-8, "Thus, a data

structure in file system unit 16 is created that contains information relating to all the

edges (and nodes) in the flow chart, including the label on each edge").

As per claim 17, the rejection of claim 1 is incorporated and further, Gunter

discloses the step of determining one or more branch destinations further comprises:

- parsing each identified branch statement to find the one or more branch

destinations designated within the branch statement (¶ 34:7-11, "The condition

"while" in code 64 is translated (i.e. replaced with) a diamond box 76. As illustrated two

branches or edges 78 and 80 are also generated relating to the "YES" (i.e. branch

statement record) and "NO" (i.e. alternative branch statement) conditions associated

with "while" construct of code 64."),

- recording the one or more branch destinations into one or more branch

destination records contained within the data structure, wherein the branch

destination records are associated with the identified branch statement record (¶

66:5-8, "Thus, a data structure in file system unit 16 is created that contains information

relating to all the edges in the flow chart, including the label on each edge"),

- recording one or more branch labels into one or more branch label

records contained within the data structure, wherein the branch label records are

associated with the branch destination records (¶ 66:5-8, "Thus, a data structure in

file system unit 16 is created that contains information relating to all the edges in the

flow chart, including the label on each edge").

As per claim 18, the rejection of claim 1 is incorporated and further, Gunter

discloses **exporting the data structure into a drawing program, wherein the data**

structure contains statement records, statement number records, destination

records, text branch label rec rds, and shape records; laying out shapes as defined in the shape records; inserting text from the statement records into the shapes; linking the shapes from the destination records; sizing the font to fit within the shapes and displaying the flowchart (¶ 35:4-16, "This DOT file is then provided (i.e. exported) to graphic drawing unit 22 via graphical interface unit 18 to determine the physical layout of the nodes and edges on display unit 14. Thus, graphic drawing unit 22 annotates graph objects with specific coordinates. Such coordinates depend, among other things, on the size of display unit 14, the number of concurrent programs being tested, the size of each program and other physical constraints of the system. Once graphic drawing unit 22 completes its graph lay-out operation, it generates an updated DOT file that includes the physical lay-out information as well. The updated DOT file is then provided to graphical interface unit 18 so that it could be displayed on display unit").

As per claim 19, this is another method version of the claimed method discussed above, in claim 13, wherein all claimed limitations have also been addressed and/or cited as set forth above. For example, see Gunter's interactive software testing system and method (¶ 32:1-38:10).

As per claim 20, this is a computer program product version of the claimed method discussed above, in claim 1, wherein all claimed limitations have also been addressed and/or cited as set forth above. For example, see Gunter's interactive software testing system and method (¶ 32:1-38:10).
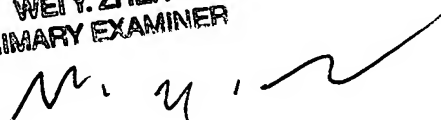
Conclusion

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Andre R. Fowlkes whose telephone number is (571) 272-3697. The examiner can normally be reached on Monday - Friday, 8:00am-4:30pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Tuan Q. Dam can be reached on (571)272-3695. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

WEI Y. ZHEN
PRIMARY EXAMINER

ARF